

Calculation of mechanism motion using discrete time approach

Lubor Zháňal¹, Petr Porteš²

Institute of Automotive Engineering, Brno University of Technology, Brno, Czech Republic

¹Corresponding author

E-mail: ¹portes@vutbr.cz, ²zhanal@vutbr.cz

Received 29 April 2018; accepted 6 May 2018

DOI <https://doi.org/10.21595/vp.2018.19941>



Copyright © 2018 Lubor Zháňal, et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract. This article describes a simple and easy to implement method for numerical computing of movement of general multibody mechanism. The method is suitable for two or three-dimensional space, rigid bodies and all types of kinematic joints. The main advantage of this method lies in the possibility of using very low discretization time step but with high computing performance due effective implementation. This approach has a positive effect to numerical stability, speed and resistance to discontinuous parameter changes. The usability of described method is verified through an experimental multibody system.

Keywords: dynamics, equations of motion, multi body, discrete time.

1. Introduction

Numerical solution of moving objects is an important theme, especially for mechanism simulations. These simulations are solved with applications called multi-body systems, where mechanisms Fig. 1 are assembled from rigid bodies, kinematic linkages, and other force interaction elements (e.g. external forces, springs, dampers, etc.). The method of solution described above is suitable for use in a computing environment of multibody systems.

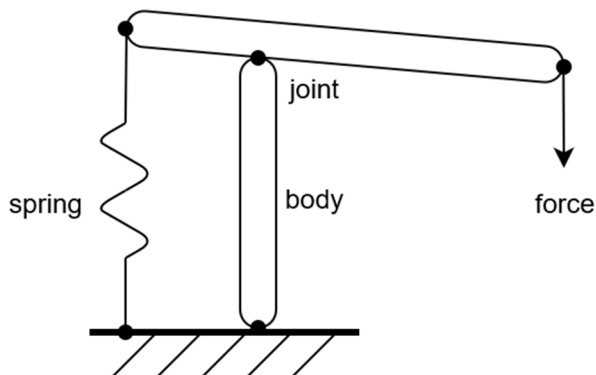


Fig. 1. Schematic mechanism made up of rigid bodies, kinematic joints, spring and outer force

For the purposes of this article, only the solution to determine the body movements if all acting forces are known is going to be discussed. This kind of body Fig. 2 can be defined as absolutely rigid, with non-zero and positive weight and with main moments of inertia. These values are related to the body's center of gravity. Furthermore, the body may contain other general points whose position is, again, determined relatively to the position of the body's center of gravity. Each common point of the body is therefore represented by their own local coordinate system (LCS), while the position and orientation of the body itself, i.e. its center of gravity, will be determined in the global coordinate system (GCS).

Each of the points can be used, for example, as an acting place of external forces or moments, references for kinematic constraints, kinematic sensors, etc.

1.1. Body movement

The position of each object in GCS can be defined using the two data structures, where the first is a vector specifying displacement of the body's center of gravity to the GCS origin, and the second is the transformation matrix for transformation from the local body space to the global space. For description of the rotation (i.e. orientation) the transformation matrix \mathbf{C} for rotation around the general axis Eq. (1) is used [1]:

$$\mathbf{C} = \begin{bmatrix} \cos\theta + e_x^2(1 - \cos\theta) & e_x e_y(1 - \cos\theta) - e_z \sin\theta & e_x e_z(1 - \cos\theta) + e_y \sin\theta \\ e_y e_x(1 - \cos\theta) + e_z \sin\theta & \cos\theta + e_y^2(1 - \cos\theta) & e_y e_z(1 - \cos\theta) - e_x \sin\theta \\ e_z e_x(1 - \cos\theta) - e_y \sin\theta & e_z e_y(1 - \cos\theta) + e_x \sin\theta & \cos\theta + e_z^2(1 - \cos\theta) \end{bmatrix}, \quad (1)$$

where e_x , e_y and e_z are the unit direction vector components of the general axis of rotation and θ is the angle of rotation.

The movement of body is then in each iteration determined by the Newton-Euler equations of motion. In the environment of the multibody system, the entire solution could be as follows:

- 1) Determining of the force and torque results in the center of gravity.
- 2) Updating the kinematic state of the center of the gravity.
- 3) Updating the kinematic state of other common points of the body.

1.2. Force and torque results of body

Points of the body Fig. 2 are indexed from zero to $N - 1$, where n is the number of points. The first point (i.e. point with zero index) is the body's center of gravity.

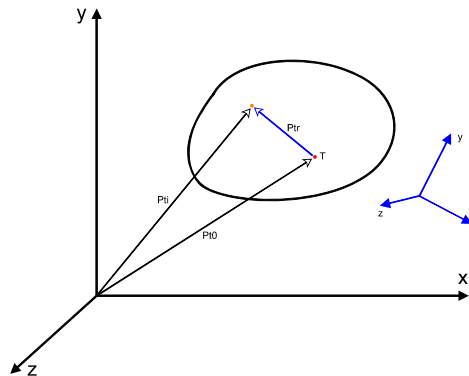


Fig. 2. A body in global and local coordinate system (highlighted)

Force and torque resultants in the body's center of gravity Eqs. (2-3) are determined as a vector sum of the forces and moments in all of the points of the body. If gravitational acceleration is acting on the body, it can also be included in the resultant equations Eq. (2):

$$\mathbf{F}_{cg} = \sum_{i=0}^{n-1} \mathbf{F}_i + m\mathbf{g}, \quad (2)$$

$$\mathbf{M}_{cg} = \sum_{i=0}^{n-1} (\mathbf{M}_i + \mathbf{P}_{tr} \times \mathbf{F}_i). \quad (3)$$

1.3. Kinematic state in the center of gravity

Body's kinematic state, or its center of the gravity, is determined by the Newton-Euler motion equations. Translational component arises from classical Newtonian motion Eq. (4), the velocity

and position is obtained by subsequent integration Eqs. (5-6):

$$\mathbf{F}_t = \frac{\mathbf{F}_{cg}}{m}, \quad (4)$$

$$\mathbf{V}_t = \int \mathbf{A}_t dt, \quad (5)$$

$$\mathbf{P}_t = \int \mathbf{V}_t dt. \quad (6)$$

Angular acceleration \mathbf{A}_r is determined on the basis of Euler's motion equation Eq. (7), where it is necessary to transform torque vector \mathbf{M}_{cg} and angular velocity vector \mathbf{V}_r using transformation matrix \mathbf{C} to the local coordinate system of the body (determined by orientation of the center of the gravity), and then the result is transformed back to the global coordinate system [2]. It is because of the correct calculation of the inertia tensor which is, in contrast to the weight, directional. Angular velocity \mathbf{V}_r is determined by simple integration of angular acceleration Eq. (8):

$$\mathbf{A}_r = \mathbf{C} \left(I^{-1} \left(\mathbf{C}' \mathbf{M}_{cg} - (\mathbf{C}' \mathbf{V}_r) \times I (\mathbf{C}' \mathbf{V}_r) \right) \right), \quad (7)$$

$$\mathbf{V}_r = \int \mathbf{A}_r dt. \quad (8)$$

To obtain the orientation of the body, which is represented by the transformation matrix of rotation around the arbitrary axis Eq. (2), is not possible to use integration of the velocity vector in a three-dimensional space, as with the translational movement (this is possible only in a plane) However, it is necessary to utilize a matrix process to compose rotations. In the form working with the discretization step Δt , the calculation then has following form: first, the angular velocity vector \mathbf{V}_r Eq. (8) is determined after the normalization of the direction of rotation axis \mathbf{e} Eq. (9) and by using the discretization step it is possible to compute the appropriate angle of rotation θ Eq. (10):

$$\mathbf{e} = \frac{\mathbf{V}_r}{|\mathbf{V}_r|}, \quad (9)$$

$$\theta = |\mathbf{V}_r| \Delta t. \quad (10)$$

Subsequently, by using the transformation matrix Eq. (1) the incremental transformation matrix is calculated, and by multiplying it with matrix of rotational position from the previous discretization step $k - 1$ the new rotational position for actual time is determined Eq. (11):

$$\mathbf{P}_{rk} = \mathbf{C}(\mathbf{e}, \theta) \mathbf{P}_{rk-1}. \quad (11)$$

1.4. Kinematic state of common points

After the kinematic state of center of the gravity is calculated, all of other kinematic states in the remaining points of the body can be easily determined. The translational position of the point in GCS is determined by transformation and addition of relative displacement vector to the translational position of the center of gravity Eq. (12). Vectors of translational velocity and accelerations are calculated as the numerical derivation of positions, or acceleration Eqs. (14-16). Orientation (i.e. "rotational position") is the product of a transformation matrix of the center of gravity and the transformation matrix in the relevant point Eq. (13). Angular acceleration and angular velocity of all points of the body are identical, so their values can be determined directly from point 0 Eqs. (15-17), which were determined when calculating the kinematic state of the body's center of gravity of the body $\forall i \in \langle 1 | n - 1 \rangle$:

$$\mathbf{P}_{t_i} = \mathbf{P}_{t_0} + \mathbf{P}_{r_0} \mathbf{P}_{tr_i}, \quad (12)$$

$$\mathbf{P}_{r_i} = \mathbf{P}_{rr_i} \mathbf{P}_{r_0}, \quad (13)$$

$$\mathbf{V}_{t_i} = \frac{\Delta \mathbf{P}_{t_i}}{\Delta t}, \quad (14)$$

$$\mathbf{V}_{r_i} = \mathbf{V}_{r_0}, \quad (15)$$

$$\mathbf{A}_{t_i} = \frac{\Delta \mathbf{V}_{t_i}}{\Delta t}, \quad (16)$$

$$\mathbf{A}_{r_i} = \mathbf{A}_{r_0}. \quad (17)$$

2. Conclusions

The method described in this article provides a simple and reliable approach to numerical solution of motion of bodies in three-dimensional space Fig. 3. Using the transformation matrix for rotation around the arbitrary axis, elegantly avoids the problems arising from the composition of rotations in space. The advantages are also easy and straightforward implementations using conventional programming languages, type C, Pascal, Matlab, etc. This method is designed to use a very small-time step in range of 10^{-5} to 10^{-7} seconds for common mechanism. In this case the numerical solution does not require any complex numerical solvers. Thanks to this, the resulting program code is short, memory transfers are minimized and all executing loops are very tight, which brings high and stable performance. Mechanisms composed of dozens of bodies, kinematic joints and other objects can be simulated in real time on an average computer.

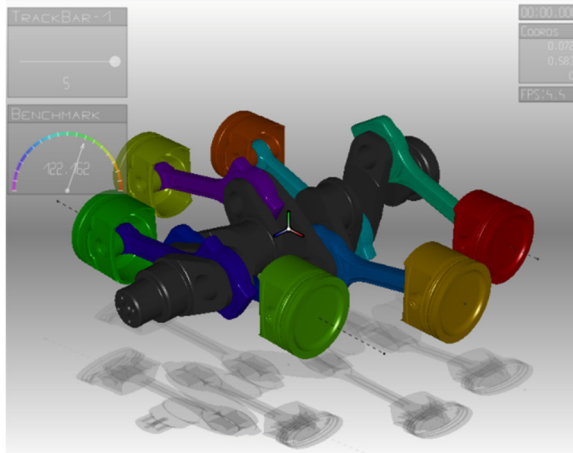


Fig. 3. Multibody system based on described method

Acknowledgement

The research leading to these results has received funding from the MEYS under National Sustainability Programme I (Project LO1202) and Specific Research Project of the Faculty of Mechanical Engineering, Brno University of Technology (FSI-S-17-4104).

References

- [1] Zháňal L. Simulation of Kinematics and Dynamics of Vehicle Mechanisms. Brno University of Technology, Brno, 2015, (in Czech).
- [2] Porteš P. Using Mathematical Vehicle Models to Analyze Measured Data. Brno University of Technology, Brno, 2015, (in Czech).